METHOD AND SYSTEM FOR GENERATING MATERIALS FOR PRESENTATION ON A NON-FRAME CAPABLE WEB BROWSER

1   BACKGROUND OF THE INVENTION

2   1.   Field of the Invention

3   The present invention relates to a system and method for presenting, over a

4   network, materials contained in a single computer file.

5   2.   Description of the Related Art

6   The Internet, as it is popularly known, has become an important

7   and useful tool for accessing a wide variety of information. One component of the

8   Internet is the World Wide Web (hereinafter, the web). In recent years the web

9   has become an increasingly popular vehicle for providing information to virtually

10   anyone with access to the Internet. Many sites (hereinafter, web sites) have been

11   established to provide information in many different forms, such as text, graphics,

12   video and audio formats over the web.

13   A typical web site includes system and application software programs

14   installed on a web server that is connected to the Internet. By connecting the web

15   server to the Internet, clients that are connected to the Internet can access the web

16   site via the web server. Usually a client is located remotely from the web server,

1   although the client and server can be at the same location. Also, a web server can

2   be connected to a private intranet, as opposed to or in addition to the public

3   Internet, in order to make a web site privately available to clients within an

4   organization.

5   A client typically accesses the web site by using a web browser. The web

6   browser is a software program which runs on the client and receives from the

7   server information formatted in a known manner. A very popular format for

8   information sent over the web from a server to a client, is the hypertext mark-up

9   language (HTML). HTML is a tag based script format, in which tags surround the

10   information to be presented. By tagging the information to be sent to a browser,

11   the browser can interpret the tags and handle the presentation of the information

12   sent from the server. It is left to the web browser, at the client, to determine the

13   specific formatting of the information, based on the tags included in the HTML

14   information sent from the server. For example, information to be displayed at a

15   client might include header information followed by a list of other information.

16   Fig. 1A shows a portion of HTML code with such a header and list. When the

17   HTML information is received by a client and interpreted by a web browser, the

18   information is displayed, as is shown in Fig. 1B, for example, in which a header 10

19   and list 12 are presented to a user at the client.

20   The HTML information sent by the server does not specify the particular

21   size, font and placement of the header 10 and the list 12. Rather, the header and

1   list information are surrounded by HTML tags which identify that information as a

2   header and a list, respectively. It is the responsibility of the web browser at the

3   client to determine how the information is presented.

4   Over the past several years, HTML-compliant web browsers have

5   proliferated at a great rate. Accordingly, a large install base of HTML-compliant

6   web browsers has been established worldwide.

7   HTML supports the use of links, which allow a web browser user to link

8   from one source of information to another, easily and rapidly. A link provides a

9   user with the means to navigate the web, and even navigate within a specific web

10   site. The server, when sending HTML information to a web browser, would

11   include a Uniform Resource Locator (URL) in information associated with the

12   link. Typically, a link would be displayed to a user in a special format, such as

13   being highlighted. The user could select that link, for example, by moving a

14   pointing device such as a computer mouse to position a cursor over the link and

15   selecting that link by pressing a button on the mouse. When a user selects a link,

16   the web browser sends a request for the corresponding URL. The URL identifies

17   a specific web resource, such as a specific web page at the web site. The server

18   receives the request for the URL, determines the information to be sent to the

19   client, accesses that information and then sends that new information to the client

20   for presentation. The client receives the new presentation information sent by the

21   server, which is typically HTML formatted information, and presents it to the user

1 via the web browser. In this manner, the World Wide Web can be navigated by

2 selecting links presented by the web browser, in which the links identify HTML

3 web pages in a web site.

4 A web site generally includes more than one web page. Typically, each

5 web page is defined in a single computer file, containing only the contents of that

6 web page. In other words, the web page is defined in a flat HTML file. When the

7 server receives a message containing a request for a URL, that URL generally

8 identifies the file for the requested web page. The server, in response to the

9 message, retrieves the file for the requested web page and transmits the contents of

10 the retrieved file to the client. In this manner, the web pages are maintained

11 separately for ease of access. However, using a separate file for each web page

12 introduces problems with maintaining multiple files. As a web site becomes larger

13 and more complex, the number of web pages generally would increase.

14 Accordingly, the number of HTML flat files becomes larger. With the increase in

15 the number of HTML flat files comes added complexity for the document creator

16 or webmaster to manage those files.

17 A solution to the problem of managing a large number of HTML flat files is

18 to include a plurality of a web site's pages in a single file. Here, each time the

19 client requests one of the plurality of pages, the server would download to the

20 client the entire single file containing the requested page as well as the other pages

21 stored in the same file but not requested. Accordingly, when a client requests any

1   of the web pages in the site, the entire file containing the requested page would be

2   sent to the requesting client, including some non-requested pages. For a very

3   small web site, a single file which contains all the web pages of the site could be

4   used. Although, while storing all the pages in a single site may be feasible, as the

5   size of the web site grows, the size of the single file would grow. Transmitting a

6   large file containing all the web pages, or even a plurality of web pages, would

7   likely cause the performance of the web site to be degraded, manifesting itself in

8   long response times at the client, and increased network traffic.

9   Accordingly, in the conventional environment described above, an HTML

10   document creator, or webmaster, is faced with either the system administration

11   headache of managing and maintaining multiple, flat HTML files, or subjecting

12   users to downloading one, potentially very large document for each request.

13   Also, a limitation of earlier versions of HTML-compliant web browsers is

14   that only a single web page could be presented to a user at any particular time.

15   Although the web provides tremendous capabilities for presenting a wide variety

16   and great depth of information, this limitation of existing web browsers to display

17   only a single page of information at a time limits the usefulness, and the power of

18   the web.

19   A solution to this problem has been to include in web browsers a feature

20   known as frames, as used in some of the NAVIGATOR products manufactured by

21   Netscape Communications Corp. A frame-capable web browser, such as

1   NAVIGATOR 3.0, allows a client to display, via the web browser, more than one

2   frame at a time. This feature allows a user to display one set of information in one

3   frame, while displaying another set of information in another frame, so that both

4   frames are displayed by the web browser at the same time. Fig. 2 shows an

5   example of a display generated by a frame-capable web browser. The display

6   shown in Fig. 2 includes four frames of information displayed at the same time. As

7   depicted in Fig. 2, a table of contents frame 20, shown on the left-hand side of the

8   display, would list items related to the table of contents of a document. A header

9   frame 22, shown in the upper portion the display, would contain information

10   identifying the information shown in a main frame 24, depicted in the middle of the

11   display of Fig. 2. The main frame 24 would contain detailed information related to

12   the item selected in the table of contents frame 20. A footer frame 26 may also be

13   included to present information at the bottom of the display, such as footnotes, or

14   a number of links, for example.

15   In this frame-based environment if a user selects one of the items contained

16   in the table of contents frame, a request for a URL is sent to the web server to

17   direct it to send new information to be displayed in the main frame relating to that

18   specific URL. For example, if the user selects an item in the table of contents

19   frame, the web browser, in response, sends to the web server a request for a URL

20   unique to that selected item. The web server receives the request for the URL

21   which might identify a separate computer file on the server. The information in

1   that file is transmitted to the web browser and the web browser displays the

2   received information in the main frame.

3   In order to manage and identify the frames in which the information is to be

4   displayed, the server requires that specific frames be identified in the HTML

5   structure of the files at the server. For example, Table 1 shows an example of

6   HTML source code necessary for creating the frames shown in Fig. 2. For

7   example, frame tags are required in the HTML source code to identify a frame.

8   Those frame tags include a frame source name (SRC) and attributes about the

9   frame such as the location for display by the web browser (e.g., row and column

10   sizes). Here for example, the table of contents frame is located in a separate file

11   named TOC.HTM. The frame tag identifies the size of the frame and location of

12   the frame. Similarly, the frame tags for the header, main and footer frames identify

13   those frames as well as the relative positions of those frames for display by the web

14   browser. Upon receiving these special frame tags, the frame-capable web browser

15   decodes those frame tags. The web browser then displays the information relating

16   to each frame in the corresponding frame. That information is displayed in areas at

17   the location and relative size specified by the frame tags.

	Table 1: HTML Source Code for A Frame Based Web Browser
1	
2	`<HTML>`
3	`<HEAD>`
4	`<TITLE>Title of Web Page With Frames</TITLE></HEAD>`
5	`<FRAMESET ROWS="20%,60%,20%">`
6	`  <FRAME SRC="header.htm">`
7	`  <FRAMESET COLS="20%,80%">`
8	`    <FRAME SRC="toc.htm">`
9	`    <FRAME SRC="main.htm">`
10	`  </FRAMESET>`
11	`  <FRAME SRC="footer.htm">`
12	`</FRAMESET>`
13	`</HTML>`
14	**File: HEADER.HTM**
15	`<HTML><BODY>`
16	Header Frame
17	`</BODY></HTML>`
18	**File: TOC.HTM**
19	`<HTML><BODY>`
20	Table of Contents Frame
21	`</BODY></HTML>`
22	**File: MAIN.HTM**
23	`<HTML><BODY>`
24	Main Frame
25	`</BODY></HTML>`
26	**File: FOOTER.HTM**
27	`<HTML><BODY>`
28	Footer Frame
29	`</BODY></HTML>`

1   Although use of frames provides another dimension of functionality for

2   display of web-based information, web sites that use frames require web browsers

3   to support the frame feature, in order to have information presented in the frames.

4   While displaying information according to a frame paradigm can be useful in

5   displaying web-based information, the size of the install base of non-frame-capable

6   web browsers limits the use of frames. That is, in order to use frames, a frame-

7   capable web browser must be installed on the client. However, because of the

8   large number of installed web browsers which are not frame-capable, use of frames

9   on web sites is often avoided, or else HTML code is included to generate both

10   frame-based and non-frame-based web pages for transmission to frame capable and

11   non-frame-capable web browsers. Creating two sets of data, one for frame-

12   capable browsers and another for non-frame capable browsers, creates difficulties

13   in managing and maintaining the redundant information.

14   Another problem with the use of presenting information in frames, is that a

15   single, unified background image cannot be presented by a web-browser that

16   presents frames of information.

17   **SUMMARY OF THE INVENTION**

18   The present invention is directed to solving the above problems. That is,

19   the present invention is directed to presenting information over the World Wide

20   Web in a manner such that information from different files or from the same file

1   can be displayed at the same time in a related manner, but without requiring a

2   frame-capable web browser.

3   An object of the present invention is to provide a system and method for

4   managing internet presentation materials in a single file format for ease of

5   administration while presenting to an internet requestor only those portions of the

6   file requested, for maximum performance.

7   Another object is to provide a system and method for presenting internet

8   information using borderless presentation areas, where the background

9   specification is decoupled from the presentation area specification.

10   Yet another object is to provide a system and method for using a dynamic

11   web page builder to generate and manage multiple simultaneous presentation

12   areas, including where one presentation area includes table of contents information

13   covering other content on a web site and where the table of contents information is

14   maintained continuously on-screen for ease of navigation through the web site.

15   Still another object of the invention is to provide a method and system for

16   generating presentation materials for transport over a network to a browser, in

17   which the browser is not capable of handling frames.

18   Yet another object of the invention is to provide a web macro, which is

19   computer-readable and embodied on a tangible medium, and which directs

20   generation of pages of presentation materials, wherein only a single instance of

1   information is maintained, yet that information is used in generating a plurality of

2   the pages of presentation material.

3   **BRIEF DESCRIPTION OF THE DRAWINGS**

4   The above objects and advantages of the present invention will become

5   more apparent by describing in detail a preferred embodiment thereof with

6   reference to the attached drawings in which:

7   **Figs. 1A and B** respectively depict HTML source code for a web page and

8   a display generated by a web browser based on that source code, as is known in

9   the prior art;

10   **Fig. 2** depicts frames produced by a frame-capable web browser, as is

11   known in the prior art;

12   **Fig. 3** is a block diagram of a client/server system for transmitting

13   presentation materials from a web-based server to a client according to the

14   invention;

15   **Fig. 4** shows presentation of materials on a web browser according to the

16   invention;

17   **Fig. 5** is a flowchart for describing a process of generating the presentation

18   materials shown in Fig. 4;

19   **Fig. 6** shows presentation of other materials on a web browser according

20   to the invention;

1   **Fig. 7** shows a display of materials with pseudo-frames, according to the

2   invention; and

3   **Fig. 8** shows a plurality of presentation areas with a uniform background

4   across all the presentation areas.

5   ## DETAILED DESCRIPTION OF THE INVENTION

6   A preferred embodiment of a method, system and article of manufacture

7   for managing internet presentation materials in a single file format according to the

8   present invention is described below in detail with reference to the accompanying

9   drawings.

10   The present invention is directed to a method, system and article of

11   manufacture for managing internet presentation materials in a single file format.

12   The invention takes advantage of properties of a dynamic HTTP application

13   software 32 program which runs on a web server 30, as shown in Fig. 3. An

14   example of a preferred dynamic HTTP application is NET.DATA manufactured by

15   IBM Corp., although, other programs capable of dynamically creating

16   internet/intranet information can be used. The dynamic HTTP application 32

17   allows information on a web page to be created or changed dynamically, while the

18   application program is running, as opposed to maintaining merely static

19   information which can be changed only by a system administrator, or similarly

1   authorized person. The dynamic HTTP application 32 operates according to a

2   web macro 34.

3   The web macro 34 includes programming statements and is an embodiment

4   of the single file format of the present invention. The web macro can be stored on

5   a tangible, computer-readable medium, such as a computer-readable disk or tape.

6   However, the web macro 34 is not limited to those media and may be recorded on

7   any media which can be read so that the programming statements are supplied to

8   the dynamic HTTP application 32. Table 2 shows an example of a portion of a

9   web macro according to the present invention. The web macro can be recorded in

10   a single file and includes two sections. The first section of the web macro is a

11   definition section, typically beginning with a statement of the form %define{

12   variable_name = {variable_value%}%}.

1	**Table 2: Web Macro with a Plurality of HTML Sections**

```
 2   %define{
 3     header = {
 4      <center>IBM Internet Yellow Pages</center>
 5     %}
 6     contents = {
 7      <b>Table of Contents</b>
 8        <ul>
 9        <li><a href="...">Purpose</a>
10        <li><a href="...">Overview</a>
11        <li>Hot <a href="...">New Technology</a>
12                        •
13                        •
14                        •
15        </ul>
16     %}
17     purpose = {
18        <b><i>Purpose</i></b>
19        <blockquote>
20                To rapidly communicate material and ...
21                <p>This internet site is updated ...
22        </blockquote>
23     %}
24     overview = {
25        <b><i>Overview</i></b>
26        <blockquote>
27                IBM's Internet Yellow Page Solution is a collection of IBM
28                software and hardware products....
29                <p>Using IBM's award-winning database, DB2, ...
30        </blockquote>
31     %}
32                        •
33                        •
34                        •
```

Table 2 (continued)

```
          •
          •
          •
  footer = {
   <a href="...">IBM home page</a> |
   <a href="...">Order</a> |
   <a href="...">Search</a> |
          •
          •
          •
   </BODY></HTML>
  %}
%}
          •
          •
          •
%HTML(purpose){
 $(header)
 $(contents)
 $(purpose)
 $(footer)
%}

%HTML(overview){
 $(header)
 $(contents)
 $(overview)
 $(footer)
%}
          •
          •
          •
  %}
```

1   Variables are defined within the %define section of the web macro. For

2 example, five variables are defined in the %define section of the web macro shown

3 in Table 2. The first variable defined in this example is the variable "header,"

4 followed by the variable "contents." Here, the value of the "contents" variable is

5 text describing a table of contents for a series of charts used in a presentation.

6 This example uses charts as the content of the web pages, merely to illustrate that

7 the value of a variable can be essentially any type of information. Here, the Table

8 of Contents includes at least three items, namely, Purpose, Overview and Hot New

9 Technology. The third variable is named "purpose." The value of the variable

10 "purpose" is text describing the purpose of the subject matter of the charts. As

11 shown in Table 2 the value of the variable "purpose" is simply some paragraphs of

12 text information. The fourth variable shown in this web macro is named

13 "overview." Here too, the value of the variable "overview" is text information.

14 The text in the "overview" variable is to be presented on the web page as part of

15 the charts for presentation at the client. The fifth variable in the web macro is

16 named "footer" and includes links for a menu bar displayed at the bottom of the

17 page.

18   The second section of the web macro is the portion of the macro for

19 generating HTML. In the example shown in Table 2 the statement %HTML{...%}

20 encompasses the HTML statements which are generated by the dynamic HTTP

21 application. In the example shown in Table 2, the HTML generated is based on

1 substitution of the contents of four variables for those variables. See

2 %HTML(purpose){...%} for example. The first variable is "header" which

3 provides header information for a header to be included in the web page to be

4 generated. The second variable is "contents." Here, the value of the variable

5 "contents," which is discussed above, is inserted in the web page. Accordingly the

6 variable "contents" in this example provides the Table of Contents to be displayed

7 on the left-hand column of the web page, as shown in Fig. 4. In a frame-based

8 system, which would require a frame-capable browser, the Table of Contents

9 would correspond to a separate frame for displaying table of contents information.

10 However, as seen here, separate frames are not required since the information can

11 be displayed in a single web page generated by the dynamic HTTP application 32.

12 The third variable in the HTML section is the variable "purpose". Here, the value

13 of the variable "purpose" is inserted into the web page by the dynamic HTTP

14 application, when the macro is interpreted. As shown in Fig. 4, the paragraphs

15 defined as the values of the variable "purpose" are included in the HTML for the

16 web page sent to the client 36. Accordingly, the web browser 38 displays those

17 paragraphs in an area that would correspond to a main frame in a frame-based

18 system. The last variable in the HTML section for this example is the variable

19 "footer". Here, a footer is produced by generating corresponding HTML tags for

20 the web page by substituting the contents of the variable "footer", resulting in a

1   footer being displayed on the web page. The footer in Fig. 4 is a menu bar 59

2   shown at the bottom of the display.

3   Multiple HTML-input sections can be included in the web macro. In the

4   example shown in Table 2, two HTML sections are defined. The first HTML

5   section defines the case where a user has selected the "Purpose" item listed in the

6   Table of Contents presented by the web browser. The result is an HTML web

7   page generated which contains a header, a table of contents, a Purpose section

8   presented in the middle portion of the display, and a footer, as shown in Fig. 4.

9   The web page shown in Fig. 4 includes a first area 50 (Table of Contents)

10   and a second area 56 (Purpose), which are associated by the list of items in the

11   first area.

12   Operation of the web macro according to the invention is described below

13   with reference to the flowchart shown in Fig. 5. When a requestor selects the

14   Purpose item 52 listed in the table of contents area 50 of the display shown in Fig.

15   4, the first HTML section of the web macro shown in Table 2 is interpreted,

16   thereby generating HTML for a web page. More specifically, when the Purpose

17   item 52 is selected by the requester (step S100), the web browser 38 generates a

18   message including a request for a URL, which is sent to the web server 30 to link

19   to the Purpose page. The web server 30 passes the request for the URL to the

20   dynamic HTTP application 32 (step S110). The dynamic HTTP application 32

21   invokes the web macro 34 and supplies the variable "purpose" to the web macro's

1 HTML sections (step S120). In this case, the first HTML section corresponds to

2 "purpose," and accordingly that portion of the HTML part of the web macro is

3 interpreted by the dynamic HTTP application 32. When the first HTML section is

4 interpreted, the value of the variable "purpose" is substituted for the variable

5 "purpose", along with values of other specified variables being substituted (step

6 S130). The result is the web page shown in Fig. 4 being generated (step S140).

7 The web page shown in Fig. 4 includes the header 58, the table of contents 50, and

8 the footer 59, and it also includes in a middle portion, or main area 56, the purpose

9 information that is defined for the "purpose" variable. Here, the value of the

10 "purpose" variable is shown in the middle portion of the display and includes the

11 word "Purpose" and two paragraphs of text. In this manner when a requestor

12 selects an item from the first area 50, corresponding to the Table of Contents in

13 this example, a request is sent to the web server 30 and ultimately to the dynamic

14 HTTP application 32 to generate another web page which includes not only the

15 information of the first area 50 (e.g., the table of contents information) but also

16 second information corresponding to information displayed in the main area 56,

17 here, the Purpose information. Once the HTML web page is generated and

18 transmitted from the web server 30 to the client 36 (step S150), the web browser

19 38 displays the HTML page as shown in Fig. 4 (step S160). Other information

20 such as header 58 and footer 59 information can also be included in the generated

21 web page, for presentation at the client by the web browser.

1   Only portions of the web macro 34 requested by the user, are interpreted.

2   Accordingly, a plurality of web page definitions can be stored and managed in a

3   single file, i.e., the web macro, yet only those portions requested by a user are

4   included in the web page generated from the web macro 34.

5   As another example, when the user selects the Overview item 53 in the

6   Table of Contents 50, a URL is generated which identifies the "overview" variable.

7   The corresponding request for the URL is transmitted to the web server 30. The

8   web server 30 passes the request for the URL to the dynamic HTTP application

9   which generates a new web page by interpreting the second HTML section of the

10   web macro 34. Here the second HTML section, which is selected when the

11   "overview" variable is identified, is interpreted since the URL identifies the variable

12   "overview." As described above, a web page is then generated having the same

13   header 58, footer 59 and Table of Contents information 50 as before, and also

14   having an Overview portion 66, which is presented in a main area of the web

15   browser display, as shown in Fig. 6. Here, the contents of the "overview" variable

16   are substituted in the HTML generated for the requested web page. Similarly, the

17   values of the other variables specified for the selected HTML section, are

18   substituted when generating the requested web page. That requested web page,

19   generated by the dynamic HTTP application, is passed to the server which

20   transmits it to the client. The client 36 receives the generated HTML for the web

21   page, passes it to the web browser 38 which displays it, thereby presenting the

1   same header 58, footer 59 and table of contents 50 presented in the previous page,

2   but now also presenting new information shown in the main area, namely, the

3   Overview information 66.

4   In the manner described above, information is defined only once, in the

5   variable definitions, and can be maintained in a single file, yet multiple web pages

6   can be generated from that single instance of the information. Accordingly, the

7   information which is used in multiple web pages can be maintained in only a single

8   location, thereby eliminating the need to maintain a separate copy of that

9   information for each web page in which it is used. Maintaining all the information

10   for the web site in a single computer file simplifies the webmaster's task of having

11   to keep track of all the separate files which comprise a web page, or web site.

12   Moreover, by only providing those portions of the web page requested by a user,

13   maintaining all the information for the web page or web site does not impact

14   performance, since only the requested portions are transmitted. Furthermore, the

15   single instance of the information stored in the web macro can be handled and

16   presented with a non-frame capable web browser. Such a non-frame capable web

17   browser can present the information, maintained in a single instance, because the

18   client receives a complete web page in response to each selection of an item in the

19   first area, rather than receiving only a frame of information. Accordingly, a web

20   page generated according to the present invention and presented at the client using

21   a non-frame capable web browser, gives the appearance of consisting of logical

1   frames, without requiring the web browser to be capable of supporting frame tags.

2   This appearance is a result of the information being dynamically updated at various

3   locations of the display, yet the web browser does not need to support frame tags

4   and must only be able to receive and present a single page of web information.

5   Accordingly, the present invention provides the capability of giving the appearance

6   of presenting information in frames, but does not require a frame-capable web

7   browser to be installed on the client in order to present that information.

8   In another embodiment of the present invention the web macro 34 can

9   include only a single HTML section, yet support generation of web pages that give

10   the appearance of including a plurality of frames. Table 3 shows an example of a

11   web macro 34 for the same web pages shown in Figs. 4 and 6, but employs only a

12   single HTML section. Here, the %define section of the web macro is the same as

13   in the embodiment discussed above, in which the "header", "contents" and "footer"

14   variables are defined the same as in the previously described embodiment.

15   However, the variables for the main area of display are included in an "if"

16   structure. Here, when the dynamic HTTP application 32 receives a request for a

17   URL from the web browser 38 which indicates a particular item selected from the

18   first area (e.g., the Table of Contents area), the variable in the "if" structure,

19   corresponding to the received URL request, is evaluated to generate the HTML

20   information to be transmitted to the client 36. For example, if a requestor selects

21   the Overview section, a request for the URL for "overview" is sent to the web

1   server 30 which pass it to the dynamic HTTP application. The dynamic HTTP

2   application executes the HTML section and evaluates the "if" statement.

3   Accordingly, the value of the variable "overview" is substituted for the

4    $(overview) statement in the HTML section of the web macro, since the argument

5   passed to the HTML section is "overview." In this manner the same HTML

6   section can be used for all the items listed in the Table of Contents area of the web

7   page.

```
Table 3:  Web Macro with Single HTML Section

%define{
 header = {
  <center>IBM Internet Yellow Pages</center>
 %}
 contents = {
  <b>Table of Contents</b>
    <ul>
     <li><a href="...">Purpose</a>
     <li><a href="...">Overview</a>
     <li>Hot <a href="...">New Technology</a>
                        •
                        •
                        •

    </ul>
 %}
 purpose = {
     <b><i>Purpose</i></b>
     <blockquote>
             To rapidly communicate material and ...
             <p>This internet site is updated ...
     </blockquote>
 %}
 overview = {
     <b><i>Overview</i></b>
     <blockquote>
             IBM's Internet Yellow Page Solution is a collection of IBM
             software and hardware products....
             <p>Using IBM's award-winning database, DB2, ...
     </blockquote>
 %}
                        •
                        •
                        •

 footer = {
  <a>IBM home page</a> |
  <a>Order</a> |
                        •
                        •
                        •
    </BODY></HTML>
```

1	Table 3 (continued)

```
                              •

                              •

                              •

%}
%HTML_INPUT{
  $(header)
  $(contents)
  %if ( "$(arg)" == "purpose" )
    $(purpose)
  %elseif ( "$(arg)" == "overview" )
    $(overview)
                              •

                              •

                              •

  $(footer)
%}
```

1   In still another embodiment the web browser 38 is controlled by the web

2   macro and web server to make the presentation of the information in the web

3   macro 34 appear even more like frames, by including borders around the various

4   logical frames, or presentation areas, of the web page, as shown in Fig. 7. More

5   specifically, the HTML generated based on the web macro 34 causes the web

6   browser to create a pseudo-border 82 around each of the presentation areas

7   defined in the HTML section, as shown in Fig. 7. Here, a border is drawn around

8   one or more of the presentation areas of the web page. The border is defined by

9   the HTML generated to produce the web page. The web page is constructed by

10   the dynamic HTTP application 32 so that the client continues to receive a full web

11   page upon each request. However, in this embodiment borders are also produced

12   to more clearly delineate the boundaries between the different presentation areas

13   displayed at the client.

14   In yet another embodiment, the web macro 34 allows a web page to be

15   generated which include a plurality of presentation areas and a background. Here,

16   the background is decoupled from the plurality of presentation areas so that the

17   background appears uniform across the presentation areas. For example, if the

18   web macro includes a background tag for pages that are generated, upon receiving

19   a request for the Purpose information discussed above, the dynamic HTTP

20   application would generate a web page having header, footer, Table of Contents

21   and Purpose presentation areas, along with a single uniform background defined in

1   the web macro. Fig. 8 shows an example of a web page created in which the

2   macro, in the HTML section for the requested information, defines a background

3   80 to be displayed. As can be seen in Fig. 8, the background 80 appears uniform

4   across all the presentation areas. This capability would not be possible were the

5   web page implemented with frame tags.

6   In still yet another embodiment, the variables are defined in one or more

7   different files. By use of "include" statements, the values of the variables can be

8   retrieved from the files in which they are stored and included in the single web

9   macro containing the HTML sections when that macro is invoked.

10   Other modifications and variations to the invention will be apparent to

11   those skilled in the art from the foregoing disclosure and teachings. Thus, while

12   only certain embodiments of the invention have been specifically described herein,

13   it will be apparent that numerous modifications may be made thereto without

14   departing from the spirit and scope of the invention. For example, the present

15   invention is described in terms of generating HTML as a language for producing

16   the web page. However, the present invention is not limited to only the HTML

17   language. Rather, the present invention includes any other language capable of

18   being dynamically generated to express materials to be presented at a client. Also,

19   the present invention is not limited to an HTTP application, but includes any other

20   dynamic application capable of dynamically substituting values of variables to

21   create presentation materials.